

AF
2



**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES**

APPELLANT'S MAIN BRIEF ON APPEAL (Amended)

APPELLANT:	Matthias Breuer	ATTORNEY DOCKET NO.:	30014200-1007
SERIAL NO.	09/921,224	GROUP ART UNIT:	2178
DATE FILED:	August 1, 2001	EXAMINER:	Kyle R. Stork
INVENTION:	"METHODS AND SYSTEMS FOR INPUTTING DATA INTO SPREADSHEET DOCUMENTS"		

Mail Stop Appeal Brief - Patents
Hon. Commissioner for Patents
P.O. Box 1450
Alexandria, Virginia 22313-1450

Dear Sir:


In accordance with the provisions of 37 C.F.R. §41.37, Appellant submits this Amended Main Brief on Appeal pursuant to the Notice of Appeal mailed on August 21, 2006 and in response to the Notice of Non-Compliant Appeal Brief dated February 20, 2007. This Amended Main Brief on Appeal replaces and corrects deficiencies in the Main Brief on Appeal filed on December 20, 2006.

Appellant filed a Pre-Appeal Brief Request for Review on August 21, 2006. In response, the reviewing panel issued a Notice of Panel Decision from Pre-Appeal Brief Review on November 24, 2006, which states that the case should proceed to the Board of Patent Appeals and Interferences. Therefore, Appellant filed the Main Brief on Appeal.

The Main Brief on Appeal was mailed within one month of the Notice of Panel Decision from Pre-Appeal Brief Review dated November 24, 2006.

The Commissioner is hereby authorized to charge any deficiency in fees associated with this communication or credit any overpayment to Deposit Account No. 19-3140. A duplicate copy of this sheet is enclosed.

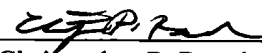
Respectfully Submitted,

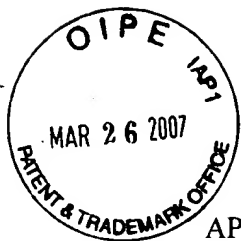
 (Reg. No. 45,034)
Christopher P. Rauch
SONNENSCHN NATH & ROSENTHAL LLP
P.O. Box #061080
Wacker Drive Station - Sears Tower
Chicago, IL 60606-1080
Telephone 312/876-2606
Customer #58328
Attorneys for Appellants



CERTIFICATE OF MAILING

I hereby certify that this original and two copies of this correspondence is being deposited with the United States Postal Service as First Class Mail in an envelope addressed to Mail Stop Appeal Brief-Patents, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on March 20, 2007.

 (Reg. No. 45,034)
Christopher P. Rauch



IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

APPELLANT'S MAIN BRIEF ON APPEAL (Amended)

APPELLANT: Matthias Breuer ATTORNEY DOCKET NO.: 30014200-1007
SERIAL NO. 09/921,224 GROUP ART UNIT: 2178
DATE FILED: August 1, 2001 EXAMINER: Kyle R. Stork
INVENTION: "METHODS AND SYSTEMS FOR INPUTTING DATA INTO
SPREADSHEET DOCUMENTS"

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, Virginia 22313-1450

Dear Sir:

In accordance with the provisions of 37 C.F.R. §41.37, Appellant submits this Main Brief on Appeal (Amended) pursuant to the Notice of Appeal mailed on August 21, 2006 in the above-identified application and in response to the Notice of Non-Compliant Appeal Brief dated February 20, 2007. This Amended Main Brief on Appeal replaces and corrects deficiencies in the Main Brief on Appeal filed on December 20, 2006.

I. REAL PARTY IN INTEREST:

The real party in interest in the present appeal is the Assignee, Sun Microsystems, Inc.. The assignment was recorded in the U.S. Patent and Trademark Office at Reel 012432, Frame 0360.

II. RELATED APPEALS AND INTERFERENCES:

Appellants are not aware of any related appeals or interferences.

III. STATUS OF CLAIMS:

Claims 1-18 are pending in the application.

Claims 1-18 are rejected due to a final rejection in the Office Action dated April 20, 2006.

Claims 1-18 are appealed.

A copy of claims 1-18 is appended hereto as the Claims Appendix.

The claim rejections are as follows:

A.) claims 1-6 and 8-16 are rejected under 35 U.S.C. §103(a) as being unpatentable over *Kanai* (U.S. Patent No. 5,339,410) (“*Kanai*”) in view of *Bhansali, et al.* (U.S. Patent No. 6,006,239) (“*Bhansali*”); and

B.) claims 7, 17, and 18 are rejected under 35 U.S.C. §103(a) as allegedly being unpatentable over *Sorge et al.* (U.S. Patent No. 6,691,281) (“*Sorge*”) in view of *Zellweger et al.* (U.S. Patent No. 6,185,582) (“*Zellweger*”) and further in view of *Bhansali, et al.*

IV. STATUS OF AMENDMENTS:

All amendments have been entered in this application.

V. SUMMARY OF CLAIMED SUBJECT MATTER:

Claims 1-18 are currently pending. Claims 1, 7, 8, 11, 17, and 18 are the only pending independent claim under consideration. A summary of independent claims 1, 7, 8, 11, 17, and 18 is presented below.

Claim 1:

Claim 1 generally relates to a method in a data processing system for modifying the content of a cell (*e.g.*, a spreadsheet cell) in a document. (Page 1, lines 15-17; claim 1). The method enables a user to input data into a document comprising cells arranged in columns and rows. (Page 9, lines 26-30). A first of the cells and a second of the cells each have an original content. (Page 8, line 30-page 9, line 1).

The original content of the first cell is overridden, without being deleted, with a first user inputted value. The first cell keeps the original content of the first cell in the first cell while the original content of the first cell is overridden. (Page 9, lines 13-19 and 26-33). The cells are recalculated based on the first user inputted value. (Page 10, lines 6-14).

After recalculating the cells based on the first user inputted value, the original content of the second cell is overridden, without being deleted, with a second user inputted value. The second cell keeps the original content of the second cell in the second cell while the original

content of the second cell is overridden. (Page 10, lines 15-25). The cells are recalculated based on the second user inputted value. (Page 10, lines 26-31).

The original content of the first cell is automatically restored based on a user input such that the second user inputted value is maintained in the second cell. (Page 10, line 32-page 11, line 16).

Claim 7:

Claim 7 generally relates to a method in a data processing system for modifying a document that includes cells arranged in rows and columns. (Page 1, lines 15-17; page 9, lines 26-30; claim 7). Each cell comprises a formula and a last result. (Page 8, lines 30-page 9, line 12).

A plurality of values for a plurality of the cells are received. (Page 9, lines 13-19; page 10, lines 15-20). The values are stored in the last result of the plurality of the cells. (Page 9, lines 30-33). The values are used during recalculation instead of the formulas. (Page 10, lines 6-9). Each of the formulas for the plurality of the cells can be restored independently of other of the plurality of cells. (Page 11, lines 2-8).

Claim 8:

Claim 8 generally relates to a data processing system in which a document having cells arranged in rows and columns can be modified. (Page 1, lines 15-17; page 7, lines 17-23; page 9, lines 26-30; claim 8). The data processing system comprises a secondary storage device including the document having cells arranged in columns and rows. (Page 7, lines 17-23). A first of the cells and a second of the cells each have an original content. (Page 8, line 30-page 9, line 1).

The data processing system includes a memory comprising a computer program that overrides, without deleting, the original content of the first cell with a first user inputted value. (Page 7, lines 24-29; page 11, lines 2-8). The first cell keeps the original content of the first cell in the first cell while the original content of the first cell is overridden. (Page 9, lines 13-19 and 26-33). The computer program recalculates the cells based on the first user inputted value. (Page 10, lines 6-14). The original content of the second cell is overridden with a second user inputted value after recalculating the cells based on the first user inputted value. The second cell keeps the original content of the second cell in the second cell while the original content of the second cell is overridden. (Page 10, lines 15-25). The cells are recalculated based on the second user inputted value. (Page 10, lines 26-31). The original content of the first cell is automatically restored based on a user input such that the second user inputted value is maintained in the

second cell. (Page 10, line 32-page 11, line 16).

A processing unit runs the computer program. (Page 7, lines 19-20).

Claim 11:

Claim 11 generally relates to a computer-readable medium containing instructions that cause a data processing system to perform a method for enabling a user to input data into a document comprising cells arranged in columns and rows. (Page 1, lines 15-17; page 7, line 30-page 8, line 5; page 9, lines 26-30; claim 11). A first of the cells and a second of the cells each have an original content. (Page 8, line 30-page 9, line 1).

The original content of the first cell is overridden, without being deleted, with a first user inputted value. The first cell keeps the original content of the first cell in the first cell while the original content of the first cell is overridden. (Page 9, lines 13-19 and 26-33). The cells are recalculated based on the first user inputted value. (Page 10, lines 6-14).

After recalculating the cells based on the first user inputted value, the original content of the second cell is overridden, without being deleted, with a second user inputted value. The second cell keeps the original content of the second cell in the second cell while the original content of the second cell is overridden. (Page 10, lines 15-25). The cells are recalculated based on the second user inputted value. (Page 10, lines 26-31).

The original content of the first cell is automatically restored based on a user input such that the second user inputted value is maintained. (Page 10, line 32-page 11, line 16).

Claim 17:

Claim 17 generally relates to a computer-readable medium containing instructions that cause a data processing system to perform a method in a data processing system comprising a document with cells arranged in rows and columns. (Page 1, lines 15-17; page 7, line 30-page 8, line 5; page 9, lines 26-30; claim 7). Each cell comprises a formula and a last result. (Page 8, lines 30-page 9, line 12).

A plurality of values for a plurality of the cells are received. (Page 9, lines 13-19; page 10, lines 15-20). The values are stored in the last result of the plurality of the cells. (Page 9, lines 30-33). The values are used during recalculation instead of the formulas. (Page 10, lines 6-9). Each of the formulas for the plurality of the cells can be restored independently of other of the plurality of cells. (Page 11, lines 2-8).

Claim 18:

Claim 18 generally relates to a computer-readable memory device encoded with a data structure with entries. (Page 7, line 30-page 8, line 5; page 8, lines 30-page 9, line 12). Each

entry reflects a cell in a spreadsheet that is recalculated by a spreadsheet program which is encoded in the memory device and which is run by a processor. (Page 7, lines 17-29). Each cell comprises: a first storage area that stores a formula, and a second storage area that stores a numerical value that temporarily overrides the formula so that the numerical value is used instead of the formula during recalculation. (Page 8, lines 30-page 9, line 12; page 10, lines 6-9).

VI. GROUND OF REJECTION TO BE REVIEWED ON APPEAL:

The grounds of rejection to be reviewed on appeal are as follows:

A.) claims 1-6 and 8-16 are rejected under 35 U.S.C. §103(a) as being unpatentable over *Kanai* (U.S. Patent No. 5,339,410) (“*Kanai*”) in view of *Bhansali, et al.* (U.S. Patent No. 6,006,239) (“*Bhansali*”); and

B.) claims 7, 17, and 18 are rejected under 35 U.S.C. §103(a) as allegedly being unpatentable over *Sorge et al.* (U.S. Patent No. 6,691,281) (“*Sorge*”) in view of *Zellweger et al.* (U.S. Patent No. 6,185,582) (“*Zellweger*”) and further in view of *Bhansali, et al.*

VII. ARGUMENT:

As set forth below, claims 1-18 are not unpatentable under 35 U.S.C. §103(a) based on the cited references. Appellant respectfully submits the Examiner's assertions are incorrect as a matter of fact and law. Thus, for the reasons set forth below, Appellant respectfully requests that this Board reverse the rejections of claims 1-18.

A.) *Kanai* in view of *Bhansali* fails to render obvious claims 1-6 and 8-16 because these references fail to disclose or suggest automatically restoring an overridden value in a cell as claimed

Kanai in view of *Bhansali* fails to disclose or suggest automatically restoring an overridden original value in a first cell after multiple recalculations, without affecting an inputted value that overrides an original value in a second cell.

Independent claims 1, 8, and 11 each claim subject matter relating to overriding without deleting the original content of a first cell with a first value and then recalculating the cells. Then, the original content of a second cell is overridden, without being deleted, with a second value and the cells are recalculated again. Then, the original content of the first cell is automatically restored, while the second (*i.e.*, overriding) value is maintained in the second cell.

The first cell keeps the original content of the first cell in the first cell while the original content of the first cell is overridden, and the second cell keeps the original content of the second cell in the second cell while the original content of the second cell is overridden. Thus, the original content of the first cell is automatically restored independently of restoring the original content of the second cell, and after multiple recalculations.

For example, the user can automatically restore an original content of a cell in any sequence, regardless of whether recalculations have been performed or whether the user has directly inputted data values into other cells. Values that have been directly inputted into other cells, even values that have been directly inputted after the directly inputted value of the cells whose original content is restored, are maintained. (Specification, page 11, lines 2-8).

This is clearly unlike *Kanai* in view of *Bhansali*, which clearly fails to disclose or suggest overriding without deleting an original content of a cell, and then automatically restoring the original content to the cell after multiple recalculations, and independently of restoring an overridden original content to a second cell.

Kanai teaches a method of bi-directionally recalculating cells. That is, recalculations of dependent cells can be performed in both directions. For example, the value in cell A may automatically change when a user inputs a new value into cell B, and the value in cell B may automatically change when a user inputs a new value into cell A. *See, e.g., Kanai* Figures 2 and

3.

The Examiner argues that *Kanai* overrides the value of a cell, without deleting the original content of the cell. Appellant disagrees. As *Kanai* clearly describes with reference to the example in *Kanai* Figures 2 and 3, *Kanai* deletes the content of a cell when a new value is written into the cell. As shown in *Kanai* Figures 2 and 3, when the value in cell X2 is changed, the value in cell Y automatically changes. That is, the value in cell Y is deleted and a new value is written into cell Y. This is because the value in cell Y is calculated based on the formula: cell Y's value = cell X1's value + cell X2's value. *Kanai* Figures 2 and 3; 3:60-4:24. Similarly, as shown in *Kanai* Figure 4, when the value in cell Y changes, the value in cell X1 automatically changes. That is, the value in cell X1 is deleted and a new value is written into cell X1. This is because the value in cell X1 depends on the value of cell Y. *Kanai* Figure 4.

The Examiner argues that these same Figures and text passages from *Kanai* teach overriding, without deleting, the original content of a cell. However, nowhere do these passages make such a teaching. Instead, they merely teach bidirectional recalculation. *Kanai*'s cells X1, X2, and Y are different cells. Their values are influenced by the values of other cells (e.g., $Y = X1 + X2$). Therefore, their values are automatically recalculated when the value in an influencing cell changes. Nowhere does *Kanai* suggest that its cell values are overridden without being deleted. Instead, *Kanai* merely teaches automatically recalculating cells, with no mention of keeping a previous value of the cell.

Bhansali also fails to disclose or suggest overriding without deleting an original content of a cell, and then automatically restoring the original content to the cell after multiple recalculations, and independently of restoring an overridden original content to a second cell.

Bhansali allows multiple users to edit a spreadsheet document by maintaining a local copy of the spreadsheet at each user's computer. *Bhansali* 3:6-10. A central copy is maintained on a centralized storage disk. Each user's computer includes a "memory change log" that keeps track of changes made by that user. *Bhansali* 7:39-45. When a user saves the user's changes, that user's memory change log is appended to a "disk change log" at the central storage disk. *Bhansali* 3:11-13. The changes of the various users are made to the centralized copy of the spreadsheet by sequentially processing each of the memory change logs that have been appended to the disk change log. When two users' edits conflict, either a user chooses the winning edit or the most recent edit is used. When a user's edit loses, an inverse of the edit is performed such that the end result is effectively that the edit never took place (e.g., a losing edit that inserted a row is reversed by subsequently deleting the row). *Bhansali* 3:14-40.

In *Bhansali*, a record of losing edits is maintained at a user's computer in a "disk undo log" (losing edits are undone in the user's local copy). When the user saves the spreadsheet, the user's disk undo log is also appended to the centralized disk change log. Thus, when the centralized copy of the spreadsheet is updated, "memory change log" edits are implemented as well as "disk undo log" edits to correct any conflicts by implementing inverse actions of the losing edits. *Bhansali* 11:45-67.

Thus, unlike Applicant's claimed invention, *Bhansali* fails to disclose or suggest overriding without deleting the original content of a cell with a user inputted value, wherein the cell keeps the original content in the cell. Instead, *Bhansali* allows a user to overwrite and delete the original value of a cell in the user's local copy. The original content of the cell can be replaced by performing an inverse action. To perform the inverse action, *Bhansali* looks to its "disk undo log" to identify the losing action and then implements the losing action's inverse. *Bhansali* 11:45-67. Thus, the original content of a cell is not maintained in the cell, but in the disk undo log. *Bhansali* deletes the original content of a cell and then restores the original content with information that it retrieves from its disk undo log. Accordingly, *Bhansali* also fails to disclose or suggest overriding without deleting the original content of a cell with a user inputted value, wherein the cell keeps the original content in the cell. This is clearly unlike Applicants' claimed invention.

Therefore, *Kanai* in view of *Bhansali* still fails to disclose or suggest overriding without deleting the original content of a cell with a user inputted value, wherein the cell keeps the original content in the cell. For at least this reason, *Kanai* in view of *Bhansali* fails to disclose or suggest claims 1, 8, and 11.

Claims 2-6, 9-10 and 12-16 depend directly or indirectly from claim 1, 8, or 11 and are therefore allowable for at least the same reasons that claims 1, 8, and 11 are allowable.

Appellant respectfully requests that the Board reverse the rejection of claims 1-6 and 8-16.

- B.) *Sorge* in view of *Zellweger* and further in view of *Bhansali* fails to render obvious claims 7 and 17 because these references fail to disclose or suggest a cell comprising a formula and a last result, wherein the value stored in the last result is used to recalculate the cell instead of the formula

Sorge in view of *Zellweger* and further in view of *Bhansali* fails to disclose or suggest automatically restoring an overridden original value in a first cell after multiple recalculations, without affecting an inputted value that overrides an original value in a second cell.

Independent claims 7 and 17 each claim a plurality of cells each comprising a formula and a last result. A plurality of values are received for the plurality of cells. The values are stored in the last result of the plurality of the cells such that the values are used during recalculation instead of the formulas and such that each of the formulas for the plurality of the cells can be restored independently of other of the plurality of cells.

Unlike Appellant's claims 7 and 17, *Sorge* in view of *Zellweger* and *Bhansali* fails to disclose or suggest a cell comprising a formula and a last result, wherein the value stored in the last result is used to recalculate the cell instead of the formula. In fact, nowhere do the cited references, taken singly or in combination, teach using a value stored in a last result of a cell to recalculate the cell instead of the cell's formula. The Examiner cites several passages from *Sorge*, alleging that *Sorge* teaches using a value stored in the last result portion of a cell instead of the cell's formula. However, none of the cited passages even relate to this. *Office Action of 4/20/2006*, pages 8-9. Instead, the cited passages merely describe that the cells of a spreadsheet-version of a spreadsheet document can be replaced (*i.e.*, deleted and overwritten) with the cells of an HTML-version of the document (*Sorge* 4:10-20) and describe that the spreadsheet-version of the document can be conveniently converted to the HTML version (*Sorge* 10:45-57). Nowhere do these or other passages from *Sorge* discuss using a last result portion of a cell to recalculate instead of the cell's formula.

As discussed above, *Sorge* maintains two versions of a spreadsheet document (a spreadsheet version and an HTML version) and replaces the cells in one version with the cells of the other version. *Sorge*, 4:10-20. When a value and formula are inserted into a spreadsheet cell from the HTML document, the value and formula overwrite the original content of the spreadsheet cell. When the cell is recalculated, *Sorge* uses the cell's formula.

Unlike claims 7 and 17, nowhere does *Sorge* discuss using a received, stored value of a cell to recalculate the cell instead of a formula of the cell. Instead, *Sorge* merely teaches replacing the original content of a cell with a value and a formula from another document. Further, *Sorge* merely teaches that a cell's formula, not a last result, is used for recalculations.

Zellweger and *Bhansali* each fail to discuss or suggest using a cell's stored value to recalculate a cell instead of the cell's formula.

Therefore, *Sorge* in view of *Zellweger* and further in view of *Bhansali* fails to disclose or suggest a cell comprising a formula and a last result, wherein the value stored in the last result is used to recalculate the cell instead of the formula. Accordingly, *Sorge* in view of *Zellweger* and further in view of *Zellweger* fails to disclose or suggest claims 7 and 17.

Appellant respectfully requests that the Board reverse the rejection of claims 7 and 17.

- C.) *Sorge* in view of *Zellweger* and further in view of *Bhansali* fails to render obvious claim 18 because these references fail to disclose or suggest temporarily overriding the content of a cell and recalculating the cell using a value instead of the cell's formula

Sorge in view of *Zellweger* and further in view of *Bhansali* fails to disclose or suggest temporarily overriding the content of a cell and recalculating the cell using a value instead of the cell's formula. Appellant's independent claim 18 claims a cell having a first storage area that stores a formula and a second storage area that stores a numerical value that temporarily overrides the formula so that the numerical value is used instead of the formula during cell recalculation.

As discussed above, *Sorge*, *Zellweger*, and *Bhansali*, taken singly or in combination, fail to disclose or suggest temporarily overriding the content of a cell and fails to disclose or suggest recalculating a cell using a value instead of the cell's formula. Instead, *Sorge* merely teaches overwriting and deleting a spreadsheet cell's contents, including its formula, with a value and formula from an HTML document. When *Sorge's* spreadsheet cell's original contents are overwritten, they are not temporarily overridden -- in fact, they no longer exist.

Further, as discussed above, *Zellweger* and *Bhansali* each fail to discuss or suggest using a cell's stored value to recalculate a cell instead of the cell's formula.

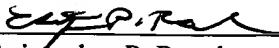
Therefore, *Sorge* in view of *Zellweger* and further in view of *Bhansali* fails to disclose or suggest claim 18.

Appellant respectfully requests that the Board reverse the rejection of claim 18.

VIII. CONCLUSION:

For the foregoing reasons, Appellants respectfully submit that the rejections posed by the Examiner are improper as a matter of law and fact. Accordingly, Appellants respectfully request the Board reverse the rejections of claims 1-18.

Respectfully submitted,

 (Reg. No. 45,034)
Christopher P. Rauch

SONNENSCHNEIN NATH & ROSENTHAL LLP
P.O. Box #061080
Wacker Drive Station - Sears Tower
Chicago, IL 60606-1080
Telephone 312/876-2606
Customer #58328
Attorneys for Appellants

CLAIMS APPENDIX

1. (Previously Presented) A method in a data processing system for enabling a user to input data into a document comprising cells arranged in columns and rows, a first of the cells and a second of the cells each having an original content, the method comprising the steps of:

overriding without deleting the original content of the first cell with a first user inputted value, the first cell keeping the original content of the first cell in the first cell while the original content of the first cell is overridden;

recalculating the cells based on the first user inputted value;

after recalculating the cells based on the first user inputted value, overriding without deleting the original content of the second cell with a second user inputted value, the second cell keeping the original content of the second cell in the second cell while the original content of the second cell is overridden;

recalculating the cells based on the second user inputted value; and

automatically restoring the original content of the first cell based on a user input such that the second user inputted value is maintained in the second cell.

2. (Original) The method of claim 1, wherein the document is a spreadsheet document and the steps of the method are performed by a spreadsheet program.

3. (Original) The method of claim 1, wherein the step of recalculating the cells based on the first user inputted value comprises automatically recalculating each cell which contains a reference to the first cell and wherein the step of recalculating the cells based on the second user inputted value comprises automatically recalculating each cell which contains a reference to the second cell.

4. (Original) The method of claim 1, further comprising the steps of:

providing to the user an option for selecting the first cell to input the first user inputted value; and

providing to the user an option for inputting the first user inputted value.

5. (Original) The method of claim 1, wherein the step of overriding the original content of the first cell with the first user inputted value further comprises the steps of:

storing the first user inputted data as a last result of a formula of the first cell;
setting a flag of the first cell to indicate that the stored last result of the first cell is valid;
and
setting a flag of each cell which references the first cell to indicate that the stored last result of each cell which references the first cell is invalid.

6. (Original) The method of claim 1, wherein each cell has a last result and a flag that indicates whether the last result is valid, and wherein the step of recalculating the cells based on the first user inputted value further comprises the steps of:

for each cell being recalculated,
determining whether the flag is set to valid;
when it is determined that the flag is not set to valid,
recalculating the last result of the cell to produce a new value;
replacing the last result with the new value such that the new value becomes the last result; and
setting the flag to valid; and
using the last result for the recalculation.

7. (Original) A method in a data processing system comprising a document with cells arranged in rows and columns, each cell comprising a formula and a last result, the method comprising the steps of:

receiving a plurality of values for a plurality of the cells; and
storing the values in the last result of the plurality of the cells such that the values are used during recalculation instead of the formulas and such that each of the formulas for the plurality of the cells can be restored independently of other of the plurality of cells.

8. (Previously Presented) A data processing system comprising:
a secondary storage device comprising a document having cells arranged in columns and rows, a first of the cells and a second of the cells each having an original content;
a memory comprising a computer program that overrides without deleting the original content of the first cell with a first user inputted value, the first cell keeping the original content of the first cell in the first cell while the original content of the first cell is overridden, recalculates the cells based on the first user inputted value, overrides without deleting the original

content of the second cell with a second user inputted value after recalculating the cells based on the first user inputted value, the second cell keeping the original content of the second cell in the second cell while the original content of the second cell is overridden, recalculates the cells based on the second user inputted value, and automatically restores the original content of the first cell based on a user input such that the second user inputted value is maintained in the second cell; and

a processing unit that runs the computer program.

9. (Original) The data processing system of claim 8, wherein the document is a spreadsheet document and the steps of the method are performed by a spreadsheet program.

10. (Original) The data processing system of claim 8, wherein each cell comprises:
a formula;
a last result of the formula; and
a flag indicating a validity of the last result.

11. (Previously Presented) A computer-readable medium containing instructions that cause a data processing system to perform a method for enabling a user to input data into a document comprising cells arranged in columns and rows, a first of the cells and a second of the cells each having an original content, the method comprising the steps of:

overriding without deleting the original content of the first cell with a first user inputted value, the first cell keeping the original content of the first cell in the first cell while the original content of the first cell is overridden;

recalculating the cells based on the first user inputted value;

after recalculating the cells based on the first user inputted value, overriding without deleting the original content of the second cell with a second user inputted value, the second cell keeping the original content of the second cell in the first cell while the original content of the second cell is overridden;

recalculating the cells based on the second user inputted value; and

automatically restoring the original content of the first cell based on a user input such that the second user inputted value is maintained.

12. (Original) The computer-readable medium of claim 11, wherein the document is a

spreadsheet document and the steps of the method are performed by a spreadsheet program.

13. (Original) The computer-readable medium of claim 11, wherein the step of recalculating the cells based on the first user inputted value comprises automatically recalculating each cell which contains a reference to the first cell and wherein the step of recalculating the cells based on the second user inputted value comprises automatically recalculating each cell which contains a reference to the second cell.

14. (Original) The computer-readable medium of claim 11, further comprising the steps of:

providing to the user an option for selecting the first cell to input the first user inputted value; and

providing to the user an option for inputting the first user inputted value.

15. (Original) The computer-readable medium of claim 11, wherein the step of overriding the original content of the first cell with the first user inputted value further comprises the steps of:

storing the first user inputted data as a last result of a formula of the first cell;

setting a flag of the first cell to indicate that the stored last result of the first cell is valid;

and

setting a flag of each cell which references the first cell to indicate that the stored last result of each cell which references the first cell is invalid.

16. (Original) The computer-readable medium of claim 11, wherein each cell has a last result and a flag that indicates whether the last result is valid, and wherein the step of recalculating the cells based on the first user inputted value further comprises the steps of:

for each cell being recalculated,

determining whether the flag is set to valid;

when it is determined that the flag is not set to valid,

recalculating the last result of the cell to produce a new value;

replacing the last result with the new value such that the new value becomes the last result; and

setting the flag to valid; and

using the last result for the recalculation.

17. (Original) A computer-readable medium containing instructions that cause a data processing system to perform a method in a data processing system comprising a document with cells arranged in rows and columns, each cell comprising a formula and a last result, the method comprising the steps of:

receiving a plurality of values for a plurality of the cells; and

storing the values in the last result of the plurality of the cells such that the values are used during recalculation instead of the formulas and such that each of the formulas for the plurality of the cells can be restored independently of other of the plurality of cells.

18. (Original) A computer-readable memory device encoded with a data structure with entries, each entry reflecting a cell in a spreadsheet that is recalculated by a spreadsheet program which is encoded in the memory device and which is run by a processor, each cell comprising:

a first storage area that stores a formula; and

a second storage area that stores a numerical value that temporarily overrides the formula so that the numerical value is used instead of the formula during recalculation.

EVIDENCE APPENDIX

Appellants do not submit additional evidence with this appeal brief and no additional evidence has been submitted during prosecution.

RELATED PROCEEDINGS APPENDIX

Appellants are not aware of any related appeals or interferences with regard to the present application.